CLAIMS

What is claimed is:

[c01] A method, comprising:

advancing instructions along a microprocessor pipeline; and

edge detecting valid instructions within the microprocessor pipeline.

[c02] A method, comprising:

fetching a bundle of instructions; and

edge detecting valid instructions within the bundle.

[c03] A method according to claim 2, further comprising shifting at least one instruction within the bundle

[c04] A method according to claim 3, further comprising rotating at least one instruction based at least in part on the number of valid instructions in the bundle.

[c05] A method according to claim 3, further comprising compressing the bundle of instructions.

[c06] A method according to claim 3, further comprising compressing the bundle of instructions for a monotonic instruction set.

[c07] A method according to claim 3, further comprising compressing the bundle of instructions based at least in part on the number of valid instructions in the bundle.

[c08] A method, comprising:

fetching a bundle of instructions having a complex instruction;

shifting at least one instruction occurring after the complex instruction; and

edge detecting the number of valid instructions occurring after the complex instruction.

[c09] A method according to claim 8, further comprising bundling instructions occurring prior to the complex instruction.

[c10] A method according to claim 8, further comprising executing instructions occurring before the complex instruction.

[c11] A method according to claim 8, further comprising bundling instructions occurring after the complex instruction.

[c12] A method according to claim 8, wherein the step of shifting the instructions comprises compressing the instructions occurring after the complex instruction.

[c13] A method according to claim 8, wherein the step of shifting the instructions comprises compressing the instructions occurring after the complex instruction for a monotonic instruction set.

[c14] A method according to claim 8, further comprising executing instructions occurring prior to the complex instruction during a first clock cycle.

[c15] A method according to claim 14, further comprising executing the complex instruction during a second clock cycle.

[c16] A method according to claim 15, wherein the step of shifting the instructions occurs while at least one of i) the instructions occurring prior to the complex instruction are executed and ii) the complex instruction is executed.

[c17] A method, comprising:

fetching an bundle of instructions having a complex instruction;

executing during a first clock cycle valid instructions occurring prior to the complex instruction;

executing the complex instruction during a second clock cycle;

shifting instructions occurring after the complex instruction during at least one of the first clock cycle and the second clock cycle;

edge detecting valid instructions occurring after the complex instruction during at least one of the first clock cycle and the second clock cycle; and

executing the valid instructions occurring after the complex instruction during a third clock cycle.